Proactive Information Retrieval via Screen Surveillance

Vuong, Tung

ACM
2017-08-07

Vuong , T , Jacucci , G & Ruotsalo , T 2017 , Proactive Information Retrieval via Screen Surveillance . in SIGIR '17 Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval . ACM , New York , pp. 1313-1316 , International ACM SIGIR Conference on Research and Development in Information Retrieval , Tokyo , Japan , 07/08/2017 . https://doi.org/10.1145/3077136.3084151

http://hdl.handle.net/10138/308885
https://doi.org/10.1145/3077136.3084151

acceptedVersion

Proactive Information Retrieval via Screen Surveillance

Tung Vuong
University of Helsinki

Giulio Jacucci
University of Helsinki

Tuukka Ruotsalo
University of Helsinki

ABSTRACT

We demonstrate proactive information retrieval via screen surveillance. A user's digital activities are continuously monitored by capturing all content on a user's screen using optical character recognition. This includes all applications and services being exploited and relies on each individual user's computer usage, such as their Web browsing, emails, instant messaging, and word processing. Topic modeling is then applied to detect the user's topical activity context to retrieve information. We demonstrate a system that proactively retrieves information from a user's activity history being observed on the screen when the user is performing unseen activities on a personal computer. We report an evaluation with ten participants that shows high user satisfaction and retrieval effectiveness. Our demonstration and experimental results show that surveillance of a user's screen can be used to build an extremely rich model of a user's digital activities across application boundaries and enable effective proactive information retrieval.

KEYWORDS

Proactive information retrieval, Screen surveillance, User modeling

1 INTRODUCTION

It has been a long-standing aim to model users based on their digital traces to personalize information to their usage context [3, 6]. A key factor in this process has been to use a history of observations regarding a user behavior to adapt search result rankings or search user interfaces.

User-activity history typically utilizes explicit user behavior information, such as queries, clickthrough data, and browsed webpages. For example, in query auto-completion, many search engines suggest the most popular completions among users based on their past history [1]. For personalized search, Teevan et al. [7] model user profiles and an operationalization from users' browsing history. Guha et al. [4] also build profiles using long-term search history for personal assistance, and Hassan et al. model search goals directly from historical observations of user behavior [5]. The use of such observational data in previous studies is usually confined to pre-defined interaction logs for particular applications and services.

In this paper, we propose an unprecedented approach, in which we use "screen surveillance" for extreme monitoring of behavioral information across boundaries of applications and systems. Our aim is to build user models from heterogeneous data based on surveillance of individual user's computer usage and utilize the models in proactive information retrieval without any direct access to predefined application interaction logs, but only extract information that is visible on the user's screen.

User activities that share a common topic usually occur on different applications needed to complete a task. For example in Figure 1, a user who works in human-resource management and is recruiting a summer trainee would need to read recruitment policy instructions, write a job advertisement, answer emails, and go through job applications, to name several activities related to the overall task spanning across a variety of applications and information items. The user would benefit from having all this information automatically retrieved for her when resuming the recruitment task.

We demonstrate a retrieval system that builds an unsupervised model of a user's topical activities from screen surveillance data. Subsequently, the model is used to detect the user's topical context from unseen user activity, and automatically retrieve relevant information. A video illustration of the system is accessible at: http://bit.ly/2rxQ06B.

We report an evaluation with 10 participants who volunteered for screen surveillance for 14 days. The results from proactive retrieval experiment using this data demonstrate high user satisfaction and retrieval effectiveness in a realistic retrieval scenario.

The rest of the paper is structured as follows. Section 2 explains the screen surveillance system, and section 3 illustrates the proactive retrieval system. Section 4 presents an evaluation in which the data from 10 participants who volunteered for screen surveillance for 14 days were used for user modeling. The participants then performed unseen activities on their computers and the system proactively retrieved information relevant to the users' activities that the users assessed according to their subjective experience of relevance. Finally, we conclude by summarizing our contributions.

2 SCREEN SURVEILLANCE SYSTEM

2.1 Surveillance Setting

Collecting everyday digital traces is a prerequisite to building a user model for a proactive information retrieval system. We used a methodology and the corresponding technology setup with the ability to collect user behavior information across boundaries of applications.

Surveillance of a computer screen reveals rich data about user behavior. The screen surveillance method has access to all interactions and all content visually presented to the user from all types of applications, including instant messaging, emails, word processing software, custom applications, operating systems, and Web browsers. It can capture content and behavior that are invisible from conventional loggers, which often monitor only Web activity and rely on clicks, queries, or webpage visits. Apart from audio, screen surveillance is able to capture every input and presentation of content that occurs between the human and the computer.

SIGIR'17, August 7–11, 2017, Shinjuku, Tokyo, Japan.

ACM ISBN 978-1-4503-5022-8/17/08…$15.00.
DOI: http://dx.doi.org/10.1145/3077136.3084151

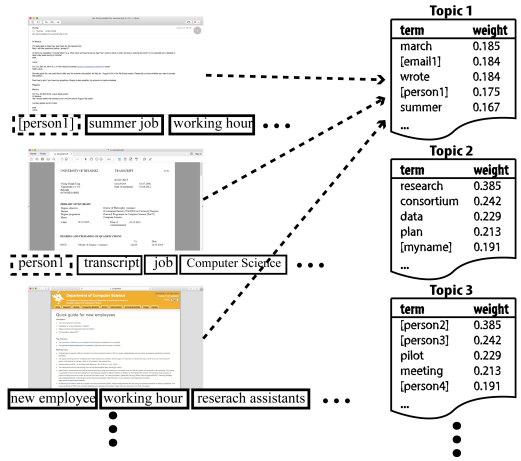

Figure 1: An example of topic modeling used to infer a user's topical activity context using a latent semantic structure on the text body of screen captures. Screen captures about the topic *"recruiting a summer trainee"* are illustrated. The screen captures of digital activities span across several applications, such as email, word processing, and Web browser applications.

2.2 Surveillance System

We use screen surveillance software to record images of active windows at five second intervals or every frame that indicates information change on the screen. Screen frames that are staying idle and constant window-switch behavior are not collected to compress the logger's allocated CPU and memory. In addition, we record operating system information that is associated with screen frames, such as the name of the active window and timestamps.

The screen surveillance system comprises of four components: Screen capture (SC) logger, Optical Character Recognition (OCR) system, keyword extraction (KE) system, and Operating System (OS) logger.

SC logger was developed in two versions. A Mac OS version was implemented by using the Core Graphics API, and a MS Windows OS version was implemented by using the Desktop App UI - both are native operating system libraries. They perform identical recording of the active window on the user's screen and save the captured screens as images. The SC also tracks mouse clicks and keystrokes. Any change in the mouse behavior or a keystroke causes the logger to activate, wait for 5-seconds until no further input is observed and commence recording the active window snapshot. Therefore, duplicate screenshots and memory overload due to constant screen capturing are avoided, and screenshots are not recorded if a computer is in idle mode.

OCR system is utilized to produce a textual representation of the content in screen capture images. We use Tesseract (version 3.04)[1] which is a prominent, accurate, cross-platform OCR engine. To obtain high precision of OCR conversion, a screen capture image was pre-processed to make the text more visible using textcleaner[2] and scaled to 500% using convert[3].

KE system extracts keyphrases and named entities from the OCR processed text. It was implemented using the Alchemy API[4].

OS logger extracts the name of the active application, the file path, and the names of people attached to the active application, which are available in many messaging applications.

All extracted data are stored in a local Lucene[5] core index for high-performance indexing and retrieval.

3 PROACTIVE RETRIEVAL SYSTEM

3.1 User Interface

In order to depict a general view of all topics resulting from the topic model's output, we designed the interface shown on the second image on the left in Figure 2. The user interface is composed of two main elements: a topic view and a document view.

The topic view visualizes an overall view of all topics from the screen surveillance database using Zoomable Circle Packing as implemented in D3.js[6]. Each big blue circle on the interface represents an individual topic. Inside the circle, descriptive labels represented as smaller white circles are grouped to characterize the topic for the user as shown on the second image on the right in Figure 2. *The document view*, shown on the right in Figure 2, visualizes the ranked list of retrieved documents. We use Collapsible Indented Tree for the document view, as implemented in D3.js.

Retrieved documents are composed from individual screen captures based on the OS log information. For instance, screen frames of the same document captured at different points in time are merged and presented as a single document. A document can be a text document from a word processing application, a web page from Web browser, a message from an instant messaging program, or a file in statistical testing tool, etc. Documents that were opened using the same application are grouped under the respective application.

3.2 Modeling and Retrieval

We utilized Latent Semantic Analysis (LSA) [2] to uncover the topical structure in the collected and OCR processed documents of screen captures. LSA learns a latent lower-dimensional representation of the input data. Each dimension in the lower-dimensional space can be interpreted as a representation of a topic and used to infer the topical activity context and retrieve associated documents and topic labels.

In the retrieval phase, the screen surveillance software runs in the background thread continuously to record screen captures of digital activities. Screen captures from the beginning of the retrieval session are constantly fed into the LSA model and the most recent screen capture is used to predict the topical activity context. Upon recognizing the topical activity context, the interface zooms in to the corresponding topic and proactively retrieves the relevant documents from the screen surveillance database by ranking the stored documents against the topic vector using cosine similarity. The retrieval process, as implemented in our system, is illustrated in Figure 2.

[1]https://github.com/tesseract-ocr/tesseract/releases/tag/3.04.01
[2]http://www.fmwconcepts.com/imagemagick/textcleaner/
[3]https://www.imagemagick.org/script/convert.php
[4]http://www.alchemyapi.com/
[5]https://lucene.apache.org/core/
[6]https://d3js.org/

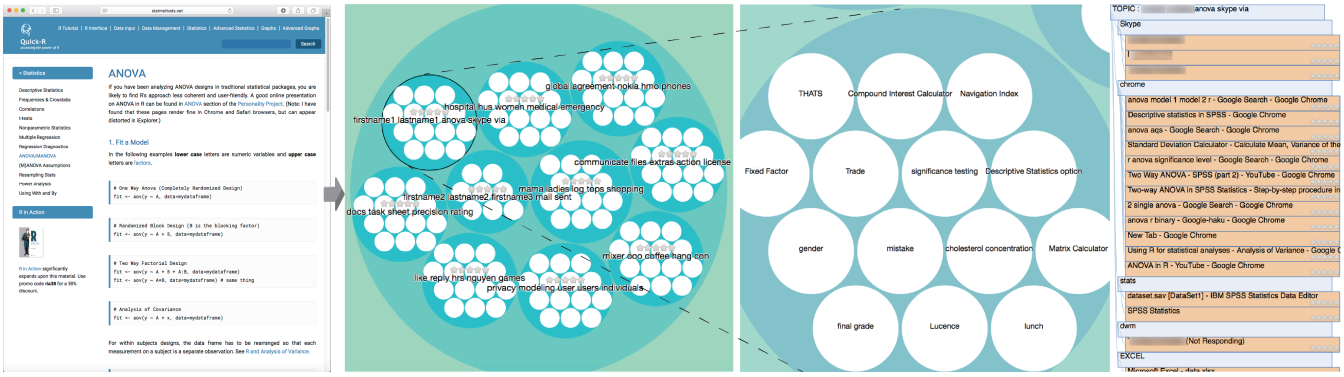

Figure 2: An illustration of the proactive information retrieval via screen surveillance. *Left*: A Web page about the ANOVA statistical test is being browsed. *Middle left*: A topic view with the detected topics. Each topic is indicated using a big blue circle. Each topic contains a set of keywords characterizing the topic, indicated by smaller white circles. The highlighted topic about Anova statistical testing is detected by the system based on surveillance of the Web page (left). *Middle right*: The system has focused on the detected topic and visualizes the keywords related to the topic. *Right*: A list of retrieved documents categorized under applications from which they were captured. The list contains documents from a variety of applications and systems. For example, Skype in which a user discussed statistical testing methods with another person, the Web browser in which the user looked up related information, the "stats" or R statistics application, Excel to run statistical tests.

4 EVALUATION

An experimental user study was conducted to study the system's retrieval performance. The evaluation consisted of two phases, which are explained below: surveillance data collection and proactive retrieval experiment.

4.1 Participants and Recruitment

Ten participants, five female and with a mean age of 28 years, were recruited to take part in the evaluation. We selected participants with higher education degrees since they would be more likely to use their personal computers for work-related tasks allowing us to collect more and more realistic data.

The participants were informed of their privacy upon joining the experiment and told that their data would be stored on an secured server, and used only for research purposes. In return for their effort, the participants were compensated with three movies tickets worth around 30 euros.

The research followed the ethical guidelines of the University of Helsinki. The research plan and informed consent form was approved by the Ethical Committee of University of Helsinki and followed the Declaration of Helsinki[7] for management of data obtained from human participants. A consent form was obtained from the users regarding the data usage policy and procedures.

4.2 Surveillance Data Collection

Prior to starting the evaluation of the retrieval system it was necessary to collect a history of user behavior data using the screen surveillance system. The screen surveillance software was installed on the participants' laptops and was set to run continuously in the background thread for 14 days. The participants were also asked to keep a daily diary of their digital activities to gain an understanding of the topics on which they were working during the surveillance period. The participants were encouraged to use their own conceptualization to obtain realistic granularity for the topics that emerged from participants' own understanding of what made a meaningful topic. The diaries resulted in 10-15 topic entries ($M = 12$).

4.3 Proactive Retrieval Experiment

After the 14-days surveillance period, the participants were called back to the laboratory to conduct the proactive retrieval experiment. We asked the participants to select six topics from their diaries on which they were still actively working or that they had worked on most recently. The participants then used a computer running screen surveillance software to perform activities related to the selected task, one at a time. The participants were explicitly advised to continue their tasks, i.e. to perform new activities dedicated to their chosen task. The participants carried out the evaluation on an isolated secured computer with Mac OS X installed. The proactive information retrieval interface was shown on an 24-inch LCD monitor.

The participants were interrupted at 30-second intervals and asked to provide relevance assessments. Every 30 seconds (up to 120 seconds), we asked the participant to look at the retrieval system, assess the relevance of the information on which the system zoomed in, and evaluate whether the retrieved documents were relevant. The system visualized the top-20 ranked documents at every iteration. We added a star-rating menu to the document and topic views to make the assessment convenient. The participants rated each document on a scale from 0 to 4. If the topic was not correctly detected after 120 seconds, the trial was marked as failed and the participant started the next trial with the next topic.

The main measures were Mean Satisfaction Score (MSS) given as an average rating of the returned information, Normalized Discounted Cumulative Gain (NDCG) and precision at N to measure the performance of the document retrieval performance. All measures were computed at every interruption point (at 30, 60, 90, and

[7] http://www.wma.net/en/20activities/10ethics/10helsinki/

Time Elapsed	*30s*	*60s*	*90s*	*120s*
TDA	0.37	0.67	0.83	0.95
MSS	3.31(0.61)	3.21 (0.67)	3.18 (0.79)	2.67 (0.98)
NDCG	0.94	0.79	0.89	0.98
P@1	1	0.78	0.9	1
P@10	0.95	0.82	0.87	0.92
P@20	0.93	0.89	0.87	0.83

Table 1: Topic detection accuracy (TDA), Mean satisfaction score (MSS), the parenthesized values indicate standard deviation, NDCG, and document precision at 1, 10, 20 in the single-trial proactive information retrieval experiment. Results are reported with respect to time (task interruption on 30 seconds interval).

120 seconds). We also report the topic detection accuracy (TDA) which indicates the cumulative accuracy of the detected topic over time.

4.4 Results

Table 1 summaries the results with respect to topic detection accuracy, mean user satisfaction, NDCG, and precision@k. In general, we observed high user satisfaction and high retrieval effectiveness after just 30 seconds, when topics were detected with an accuracy of 0.37. Most of the topics were correctly detected at 90 seconds, with the corresponding accuracy being 0.83. The satisfaction, NDCG, and P@10 were 3.18, 0.89, and 0.87 respectively.

The topic detection accuracy was low in the beginning of the session, but rapidly increased to over 0.8 at 90 seconds. The mean satisfaction score was high throughout the session, indicating that the retrieved information was of high quality if the topic was correctly detected.

The results also show constantly high precision and NDCG, but a slight drop after 60 seconds. This indicates that a small portion of topics were harder to detect and required more evidence for the topic model to converge, which took more time. This is also visible in the mean satisfaction score, which decreases slightly after 90 seconds.

5 DISCUSSION AND CONCLUSIONS

In this paper, we exploited screen surveillance for user modeling and proactive information retrieval. We studied what can be learned from the user without any control over the input or data structure, but only by continuous screen surveillance. This was operationalized by monitoring user activity using screen capturing and optical character recognition across all applications used on a computer.

We presented a system that records the computer screen and represents the content using a topic model. The model was then used to infer the topical context from natural computer usage, visualize topics for the user, and proactively retrieve topically relevant information by observing unseen user interactions.

We also reported a 2-week 24/7 surveillance experiment and single-trial information retrieval experiment using this data. The results show high user satisfaction and over 90% retrieval effectiveness using several standard measures.

This work demonstrates that using a simple, but rich signal of user activity via screen surveillance can be highly effective for inferring diverse and subtle human interests. The resulting models can be effectively applied to proactive information retrieval without reliance on any explicit user input, or underlying data or input structure.

The implications of this demonstration include the capability to identify topics representing users' activities and interests across applications and services without needing a system of service specific access or even any prior knowledge about the users. The implication is that comprehensive user modeling and proactive search can be carried out across system boundaries, as opposed to the current practice of utilizing partial data or views of users' activities. This can help to avoid cold start problems in new services and achieve better user model performance [8].

The demonstration also provocatively suggests that user modeling and retrieval results can be obtained with processes on end-user devices with ownership and control by users. This has important implications that contribute to promoting user centered personal data management in which user model data is not inaccessible and locked inside silos of search engines or other content distribution services but can be owned and accessed under the user's control.

Future work should focus on temporal models that can account for temporally evolving topics as well as hierarchical modeling that could automatically fit the number of topics to a larger variety of micro and macro tasks. Experimentation involving topic detection and retrieval accuracy should also validate the models beyond a controlled laboratory experiment.

6 ACKNOWLEDGEMENTS

This research was partially funded by TEKES (Re:Know) and the Academy of Finland (278090, 305739).

REFERENCES

[1] Ziv Bar-Yossef and Naama Kraus. 2011. Context-sensitive Query Autocompletion. In *Proc. WWW*. ACM, New York, NY, USA, 107–116.
[2] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *JASIST* 41, 6 (1990), 391–407.
[3] Gerhard Fischer. 2001. User Modeling in Human–Computer Interaction. *UMUAI* 11, 1-2 (March 2001), 65–86.
[4] Ramanathan Guha, Vineet Gupta, Vivek Raghunathan, and Ramakrishnan Srikant. 2015. User Modeling for a Personal Assistant. In *Proc. WSDM*. ACM, New York, NY, USA, 275–284.
[5] Ahmed Hassan, Rosie Jones, and Kristina Lisa Klinkner. 2010. Beyond DCG: User Behavior As a Predictor of a Successful Search. In *Proc. WSDM*. ACM, New York, NY, USA, 221–230.
[6] Tuukka Ruotsalo, Giulio Jacucci, Petri Myllymäki, and Samuel Kaski. 2014. Interactive Intent Modeling: Information Discovery Beyond Search. *Commun. ACM* 58, 1 (Dec. 2014), 86–92.
[7] Jaime Teevan, Susan T. Dumais, and Eric Horvitz. 2005. Personalizing Search via Automated Analysis of Interests and Activities. In *Proc. SIGIR*. ACM, New York, NY, USA, 449–456.
[8] Chirayu Wongchokprasitti, Jaakko Peltonen, Tuukka Ruotsalo, Payel Bandyopadhyay, Giulio Jacucci, and Peter Brusilovsky. 2015. User model in a box: Cross-system user model transfer for resolving cold start problems. In *Proc. UMAP*. Springer, 289–301.